

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai acuan untuk memahami perkembangan kajian yang berkaitan dengan perancangan basis data, normalisasi, integrasi data, pengembangan aplikasi berbasis online, metode *prototyping*, serta pengujian penerimaan pengguna. Kajian terdahulu juga digunakan untuk mengetahui metode, hasil, dan keterbatasan penelitian sebelumnya sehingga posisi penelitian dapat ditentukan secara lebih jelas.

Berdasarkan Tabel 2.1, beberapa penelitian terdahulu membahas perancangan basis data menggunakan metode *Database System Development Life Cycle* (DBSDLC). Metode tersebut diterapkan melalui beberapa tahapan, mulai dari analisis kebutuhan, perancangan konseptual, perancangan logis, perancangan fisik, implementasi, hingga pengujian [1], [2], [23]. Berdasarkan hasil kajian tersebut, DBSDLC dapat digunakan sebagai pendekatan yang sistematis dalam merancang basis data sesuai dengan kebutuhan organisasi.

Selain perancangan basis data, beberapa penelitian juga membahas penerapan normalisasi untuk mengurangi redundansi, meningkatkan konsistensi, serta menjaga integritas data [3], [4], [12]. Normalisasi diperlukan dalam perancangan basis data karena struktur tabel yang kurang tepat dapat menimbulkan duplikasi data, anomali, serta hambatan dalam pengelolaan informasi.

Penelitian lain membahas pemanfaatan basis data terintegrasi dan *dashboard Business Intelligence* sebagai media penyajian informasi [8], [10], [11], [24]. Melalui *dashboard*, data dapat disajikan dalam bentuk tabel, grafik, dan indikator sehingga informasi lebih mudah dipahami serta dapat mendukung proses *monitoring* dan pengambilan keputusan.

Selain itu, terdapat penelitian yang menggunakan metode *prototyping* dalam pembangunan aplikasi berbasis *website*. Metode *prototyping* memungkinkan rancangan awal aplikasi dievaluasi terlebih dahulu oleh pengguna sebelum

dikembangkan menjadi sistem akhir [31], [33]. Pendekatan ini dapat membantu proses pengembangan aplikasi agar lebih sesuai dengan kebutuhan pengguna karena pengguna dapat memberikan masukan pada tahap awal pengembangan.

Pengujian penerimaan pengguna juga menjadi bagian penting dalam pengembangan aplikasi. Beberapa penelitian menggunakan *User Acceptance Test* (UAT) untuk mengetahui apakah sistem yang dibangun telah sesuai dengan kebutuhan dan harapan pengguna [31], [32], [34]. UAT digunakan untuk menilai aspek kemudahan penggunaan, kesesuaian fitur, kejelasan informasi, serta manfaat sistem dalam mendukung pekerjaan pengguna.

Berdasarkan penelitian terdahulu, perancangan basis data, normalisasi, integrasi data, pengembangan aplikasi, dan pengujian penerimaan pengguna memiliki peran penting dalam mendukung pengelolaan informasi organisasi. Namun, kajian yang menggabungkan analisis sistem berjalan, perancangan basis data terintegrasi menggunakan DBSDLC, pembangunan aplikasi penjualan BBM berbasis online menggunakan metode *prototyping*, serta pengukuran kepuasan pengguna menggunakan UAT pada lingkungan operasional SPBU masih terbatas. Berdasarkan kesenjangan tersebut, penelitian ini diarahkan pada perancangan basis data terintegrasi dan pembangunan aplikasi penjualan BBM berbasis online pada PT. Indra Perkasa SPBU.

Tabel 2. 1 Penelitian Terdahulu

No	Peneliti	Tahun	Metode	Topik Penelitian	Hasil Penelitian
1	Bramastha, Wicaksono, dan Herlambang	2023	DBSDLC	Pengembangan database e-learning PT XYZ	Database berhasil dirancang menggunakan DBSDLC, MySQL, dan hasil pengujian menunjukkan sistem sesuai dengan kebutuhan [1].

2	Rahmawati, Sari, Shakti, dan Yomura	2023	DBSDLC	Perancangan database sistem bisnis Wedang Uwuh	DBSDLC menghasilkan rancangan basis data konseptual, logis, dan fisik untuk mendukung proses bisnis [2].
3	Hardini, Agarwal, Apriani, Widjaya, Setiawaty, dan Nurasiah	2025	Normalisasi 1NF–3NF	Normalisasi database untuk efisiensi penyimpanan	Normalisasi dapat mengurangi redundansi, meningkatkan integritas data, dan memperbaiki struktur tabel [3].
4	Taipalus	2025	Experimental Analysis	Pengaruh logical database design terhadap performa database	Desain database logis memengaruhi ukuran database, kompleksitas query, performa, dan konsumsi energi [4].
5	Purnamasari dan Kunang	2014	Business Intelligence	Dashboard BI untuk penentuan jumlah kelas	Dashboard BI membantu mengolah data historis untuk mendukung pengambilan keputusan jumlah kelas [8].
6	Maukar dan Wardhana	2026	ETL dan BI Development	Integrated BI dengan centralized data architecture	Database terpusat dan proses ETL membantu mengurangi inkonsistensi serta mempercepat monitoring data [10].

7	Kesuma, Riza, Nursalman, dan Firdaus	2026	ETL dan Power BI	Visualisasi data penelitian menggunakan Power BI	Dashboard Power BI mendukung monitoring, evaluasi, transparansi, dan keputusan berbasis data [11].
8	Maulana dan Samopa	2025	CDM, LDM, PDM, Normalisasi	Normalisasi untuk konsistensi database job offering	Normalisasi memperbaiki pelanggaran 3NF dan meningkatkan konsistensi rancangan database [12].
No	Peneliti	Tahun	Metode	Topik Penelitian	Hasil Penelitian
9	Harms	2012	Database Development Lifecycle	Pengembangan database berdasarkan kebutuhan klien	Database perlu dikembangkan berdasarkan kebutuhan nyata agar sesuai dengan spesifikasi pengguna [23].
10	Azuddin, Yusup, Setiyowati, Wibowo, Suwita, Basuki, dan Astuti	2025	ER Model, Normalisasi, SQL	Relational database untuk Academic Information System	Database relasional meningkatkan akses, integritas, konsistensi, dan efisiensi pengelolaan data [24].
11	Aprilia dan Dermawan	2024	Prototype dan UAT	Pengembangan sistem informasi Point of Sales berbasis website	Sistem POS berbasis website dikembangkan menggunakan metode prototype dan diuji menggunakan UAT serta Black Box Testing [31].

12	Jimmy dan Suwitno	2023	User Acceptance Test	Sistem informasi penjualan online berbasis web	UAT digunakan untuk mengetahui penerimaan pengguna terhadap sistem penjualan online berbasis web [32].
13	Kustanto, Ramadhan, dan Noe'man	2024	Prototype	Penerapan metode prototype dalam perancangan aplikasi	Metode prototype membantu pengembangan sistem melalui pembuatan rancangan awal yang dapat dievaluasi dan diperbaiki sesuai kebutuhan pengguna [33].
14	Yakub, Daniawan, Wijaya, dan Damayanti	2024	User Acceptance Testing	Sistem informasi e-commerce berbasis website	UAT digunakan untuk mengukur kesesuaian sistem berbasis website terhadap kebutuhan pengguna [34].

2.2 Teori yang berkaitan

Teori yang berkaitan digunakan sebagai dasar konseptual dalam mendukung perancangan basis data, pengelolaan data, integrasi data, serta penyajian informasi melalui *dashboard* visualisasi data. Teori yang digunakan meliputi basis data, basis data relasional, sistem manajemen basis data, normalisasi, integrasi data, *Entity Relationship Diagram* (ERD), dan *dashboard* visualisasi data.

2.2.1 DBSDLC

Database System Development Life Cycle (DBSDLC) merupakan metodologi yang digunakan sebagai pedoman dalam proses perencanaan, perancangan, pengembangan, implementasi, dan pengelolaan basis data secara sistematis. Metodologi ini bertujuan untuk menghasilkan basis data yang mampu memenuhi kebutuhan pengguna, memiliki struktur yang baik, serta menjaga konsistensi dan integritas data. Dengan adanya tahapan yang terstruktur, proses pengembangan basis data dapat dilakukan secara lebih efektif sehingga mampu meminimalkan kesalahan dalam perancangan maupun implementasi basis data [35].

Menurut Connolly dan Begg (2015), *Database System Development Life Cycle* (DBSDLC) terdiri atas serangkaian tahapan yang saling berkaitan, mulai dari proses identifikasi kebutuhan hingga implementasi basis data. Setiap tahapan memiliki tujuan tertentu dalam menghasilkan rancangan basis data yang sesuai dengan kebutuhan organisasi. Pendekatan ini membantu pengembang dalam mendokumentasikan kebutuhan sistem, merancang struktur data yang terorganisasi, mengurangi redundansi data, serta meningkatkan integritas dan konsistensi data yang disimpan dalam basis data [35].

DBSDLC memberikan pendekatan yang sistematis dalam proses pengembangan basis data sehingga setiap tahapan dapat dilakukan secara bertahap dan terstruktur. Melalui pendekatan tersebut, proses analisis kebutuhan, perancangan model data, implementasi struktur basis data, hingga proses pengelolaan data dapat dilakukan secara lebih terarah sehingga menghasilkan basis data yang efisien, mudah dipelihara, dan mampu mendukung kebutuhan sistem informasi [35].

2.2.2 ERD

Entity Relationship Diagram (ERD) merupakan model konseptual yang digunakan untuk menggambarkan struktur basis data melalui entitas, atribut, serta hubungan antar entitas. ERD berfungsi sebagai representasi visual yang membantu perancang sistem dalam memahami kebutuhan data sebelum basis data diimplementasikan ke dalam *Database Management System* (DBMS). Dengan menggunakan ERD, struktur basis data dapat dirancang secara lebih sistematis

sehingga mampu meminimalkan redundansi data serta menjaga konsistensi hubungan antar data [35].

Menurut Connolly dan Begg [35], ERD terdiri atas beberapa komponen utama, yaitu entitas, atribut, dan hubungan. Entitas merupakan objek atau himpunan objek yang memiliki karakteristik tertentu dan dapat dibedakan dari objek lainnya. Atribut merupakan karakteristik yang dimiliki oleh suatu entitas untuk mendeskripsikan informasi yang disimpan, sedangkan hubungan menunjukkan keterkaitan antara satu entitas dengan entitas lainnya.

Dalam ERD, setiap entitas umumnya memiliki *primary key* yang berfungsi sebagai identitas unik untuk membedakan setiap data pada suatu tabel. Selain itu, terdapat *foreign key* yang digunakan untuk menghubungkan satu tabel dengan tabel lainnya. Hubungan antar entitas juga dijelaskan melalui kardinalitas yang menunjukkan jumlah keterlibatan entitas dalam suatu relasi, seperti *one-to-one*, *one-to-many*, dan *many-to-many* [35].

2.2.3 Normalisasi

Normalisasi basis data merupakan proses penyusunan struktur tabel dalam basis data relasional agar data dapat disimpan secara lebih terorganisasi, konsisten, dan efisien. Proses ini dilakukan dengan membagi data ke dalam beberapa tabel berdasarkan ketergantungan antar atribut. Normalisasi bertujuan untuk mengurangi duplikasi data, menjaga integritas data, serta meminimalkan risiko anomali pada proses penambahan, perubahan, dan penghapusan data [35].

Dalam basis data relasional, bentuk normal yang umum digunakan meliputi *First Normal Form* (1NF), *Second Normal Form* (2NF), dan *Third Normal Form* (3NF). *First Normal Form* (1NF) terpenuhi apabila setiap atribut memiliki nilai atomik dan tidak terdapat kelompok data berulang. *Second Normal Form* (2NF) terpenuhi apabila tabel telah memenuhi 1NF dan setiap atribut non-kunci bergantung sepenuhnya pada *primary key*. *Third Normal Form* (3NF) terpenuhi apabila tabel telah memenuhi 2NF dan tidak terdapat ketergantungan transitif antar atribut non-kunci [35].

2.2.4 UML

Unified Modeling Language (UML) merupakan bahasa pemodelan visual yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan sistem perangkat lunak. UML membantu proses analisis dan perancangan sistem dengan menyediakan notasi standar yang dapat digunakan untuk menjelaskan kebutuhan sistem, alur aktivitas, struktur sistem, serta hubungan antar komponen di dalam sistem [36].

Dalam analisis dan perancangan sistem, UML dapat digunakan melalui beberapa jenis diagram, seperti *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*. *Use Case Diagram* digunakan untuk menggambarkan hubungan antara aktor dengan fungsi utama yang tersedia dalam sistem. *Activity Diagram* digunakan untuk menggambarkan alur aktivitas atau proses bisnis yang terjadi pada sistem, termasuk aktivitas yang dilakukan oleh aktor dan respons dari sistem. *Class Diagram* digunakan untuk menggambarkan struktur kelas, atribut, metode, serta hubungan antar kelas yang membentuk sistem [36].

Penggunaan UML dapat membantu perancang sistem dalam memahami kebutuhan pengguna dan menerjemahkannya ke dalam rancangan sistem yang lebih terstruktur. Dengan adanya pemodelan UML, proses perancangan sistem dapat terdokumentasi dengan lebih jelas sehingga mempermudah pengembangan, evaluasi, dan komunikasi antara pihak yang terlibat dalam pengembangan sistem [36].

2.2.5 Basis Data

Basis data merupakan kumpulan data yang saling berhubungan dan disimpan secara terstruktur agar dapat diakses, dikelola, serta digunakan sesuai dengan kebutuhan organisasi. Penggunaan basis data diperlukan untuk mendukung proses penyimpanan, pencarian, pembaruan, dan pengolahan data secara lebih sistematis [1], [2].

Dalam lingkungan organisasi, basis data berperan sebagai pusat penyimpanan informasi yang mendukung aktivitas operasional dan proses pengambilan keputusan. Struktur basis data yang baik dapat membantu mengurangi duplikasi

data, menjaga konsistensi informasi, serta meningkatkan efisiensi dalam pengelolaan data [24], [27].

2.2.6 SQL

Structured Query Language (SQL) merupakan bahasa yang digunakan untuk mengelola dan memanipulasi data pada basis data relasional. SQL dapat digunakan untuk membuat struktur basis data, memasukkan data, memperbarui data, menghapus data, serta mengambil data sesuai dengan kebutuhan pengguna [1], [24].

Dalam pengelolaan basis data, SQL memiliki peran penting karena mendukung perintah *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). DDL digunakan untuk mendefinisikan struktur basis data, sedangkan DML digunakan untuk melakukan proses manipulasi data, seperti SELECT, INSERT, UPDATE, dan DELETE [1].

2.2.7 Basis Data Relasional

Basis data relasional merupakan model penyimpanan data yang menggunakan tabel sebagai struktur utama. Setiap tabel terdiri atas baris dan kolom, sedangkan hubungan antar tabel dibangun melalui *primary key* dan *foreign key*. Model relasional banyak digunakan karena mampu mendukung pengelolaan data secara terstruktur, konsisten, dan mudah dikembangkan [24].

Dalam perancangan basis data relasional, hubungan antar entitas perlu disusun secara tepat agar data yang tersimpan dapat merepresentasikan kebutuhan sistem. Penerapan model relasional juga mendukung proses normalisasi sehingga redundansi dapat dikurangi dan integritas data dapat dipertahankan [3], [12], [24].

2.2.8 Sistem Manajemen Basis Data (DBMS)

Sistem Manajemen Basis Data atau *Database Management System* (DBMS) merupakan perangkat lunak yang digunakan untuk mengelola basis data. DBMS menyediakan fasilitas untuk menyimpan, mengambil, mengubah, menghapus, mengamankan, serta mengatur akses terhadap data yang tersimpan dalam basis data [1], [2].

Penggunaan DBMS membantu proses pengelolaan data agar lebih terpusat dan terkendali. Melalui DBMS, pengaturan keamanan, integritas, pemeliharaan, serta akses data dapat dilakukan secara lebih efektif sesuai kebutuhan organisasi [1], [24].

2.2.9 Normalisasi Basis Data

Normalisasi basis data dilakukan untuk menyusun struktur tabel agar data dapat disimpan secara lebih efisien dan tidak menimbulkan redundansi. Proses normalisasi diterapkan secara bertahap melalui bentuk normal tertentu, seperti *First Normal Form* (1NF), *Second Normal Form* (2NF), dan *Third Normal Form* (3NF) [3], [12].

Penerapan normalisasi dapat membantu mengurangi duplikasi data, mencegah anomali, serta meningkatkan konsistensi dan integritas data. Struktur tabel yang telah dinormalisasi mempermudah pengelolaan basis data karena setiap data ditempatkan berdasarkan ketergantungan dan relasinya [3], [4], [12].

2.2.10 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk menggambarkan hubungan antar entitas dalam suatu basis data. ERD membantu menjelaskan entitas, atribut, *primary key*, *foreign key*, serta relasi antar tabel sebelum rancangan basis data diimplementasikan ke dalam DBMS [2], [24].

Penggunaan ERD dapat mempermudah proses perancangan basis data karena hubungan antar data dapat divisualisasikan secara lebih jelas. Dengan adanya ERD, struktur basis data dapat disesuaikan dengan kebutuhan sistem dan aturan bisnis yang berlaku dalam organisasi [2], [24].

2.2.11 Integrasi Data

Integrasi data dilakukan dengan menggabungkan data dari berbagai sumber ke dalam satu lingkungan penyimpanan agar dapat dikelola secara terpadu. Proses integrasi diperlukan ketika informasi organisasi berasal dari beberapa aktivitas atau bagian yang berbeda [10].

Melalui integrasi data, informasi dapat disajikan secara lebih konsisten karena data yang digunakan berasal dari sumber penyimpanan yang terpusat. Basis data terintegrasi juga dapat mendukung proses pelaporan, *monitoring*, dan analisis informasi secara lebih efektif [10], [27].

2.2.12 Aplikasi Berbasis Online

Aplikasi berbasis online merupakan aplikasi yang dapat diakses melalui jaringan internet dengan menggunakan perangkat yang terhubung ke sistem. Aplikasi ini memungkinkan proses pengelolaan data dilakukan secara lebih fleksibel karena pengguna dapat mengakses informasi melalui *browser* atau perangkat yang mendukung layanan internet [31], [32].

Dalam penelitian ini, aplikasi berbasis online digunakan untuk mendukung pengelolaan data penjualan BBM pada PT. Indra Perkasa SPBU. Aplikasi tersebut dirancang agar data operasional, seperti data penjualan, stok BBM, pendapatan shift, biaya, delta bulanan, dan laba dapat dikelola melalui sistem yang terhubung dengan basis data. Dengan adanya aplikasi berbasis online, proses pengelolaan dan penyajian informasi dapat dilakukan secara lebih terstruktur.

2.2.13 Class Diagram

Class Diagram merupakan salah satu diagram UML yang digunakan untuk menggambarkan struktur sistem dalam bentuk kelas, atribut, *method*, dan hubungan antar kelas. Diagram ini membantu menjelaskan bagaimana komponen dalam sistem saling berhubungan serta bagaimana data dan fungsi pada sistem disusun secara terstruktur [25].

Dalam pengembangan sistem informasi, *Class Diagram* digunakan untuk memberikan gambaran mengenai struktur logis sistem sebelum sistem diimplementasikan. Melalui *Class Diagram*, entitas utama, atribut data, serta relasi antar bagian sistem dapat terlihat dengan lebih jelas. Pada penelitian ini, *Class Diagram* digunakan untuk menggambarkan struktur aplikasi penjualan BBM berbasis online, seperti kelas pengguna, penjualan harian, stok BBM, pendapatan shift, biaya, laba kotor, delta bulanan, laba bersih, dan laporan.

2.2.14 Use Case Diagram

Use Case Diagram merupakan salah satu diagram UML yang digunakan untuk menggambarkan hubungan antara aktor dan fungsi yang terdapat dalam sistem. Diagram ini menunjukkan siapa saja pengguna sistem serta fitur apa saja yang dapat diakses oleh masing-masing pengguna [25].

Use Case Diagram digunakan untuk memperjelas kebutuhan fungsional sistem sebelum aplikasi dikembangkan. Melalui diagram ini, peran pengguna dan batasan akses terhadap fitur sistem dapat digambarkan secara lebih terstruktur. Pada penelitian ini, *Use Case Diagram* digunakan untuk menggambarkan interaksi antara Admin dan Supervisor dengan aplikasi penjualan BBM berbasis online.

2.2.15 Activity Diagram

Activity Diagram merupakan diagram UML yang digunakan untuk menggambarkan alur aktivitas atau proses kerja dalam suatu sistem. Diagram ini menunjukkan urutan aktivitas, percabangan, keputusan, serta aliran proses dari awal hingga akhir [25].

Dalam perancangan sistem, *Activity Diagram* dapat membantu menjelaskan alur kerja pengguna ketika menjalankan suatu fitur. Pada penelitian ini, *Activity Diagram* digunakan untuk menggambarkan alur proses pada aplikasi penjualan BBM berbasis online, seperti proses login, *import* data, pengelolaan data, melihat *dashboard*, dan penyusunan laporan.

2.3 Framework/Algoritma dan Metode

Framework atau metode yang digunakan dalam penelitian ini meliputi *Database System Development Life Cycle* (DBSDLC), metode *prototyping*, dan *User Acceptance Test* (UAT). Ketiga metode tersebut digunakan sesuai dengan fokus penelitian, yaitu analisis sistem berjalan dan perancangan basis data terintegrasi, pembangunan aplikasi penjualan BBM berbasis online, serta pengukuran kepuasan pengguna terhadap aplikasi yang dikembangkan.

DBSDLC digunakan sebagai pendekatan dalam perancangan basis data karena penelitian ini berfokus pada pengelolaan data operasional yang membutuhkan

struktur penyimpanan yang terintegrasi. Metode ini digunakan untuk mendukung tahapan analisis kebutuhan, perancangan basis data, implementasi basis data, dan pengujian basis data [1], [2], [23].

Metode *prototyping* digunakan dalam pembangunan aplikasi penjualan BBM berbasis online. Metode ini dipilih karena rancangan awal aplikasi perlu dievaluasi terlebih dahulu agar tampilan, alur, dan fitur yang dibangun sesuai dengan kebutuhan pengguna. Melalui metode *prototyping*, proses pengembangan aplikasi dapat dilakukan secara bertahap berdasarkan masukan dari pengguna [31], [33].

User Acceptance Test (UAT) digunakan untuk mengukur tingkat kepuasan pengguna terhadap aplikasi yang telah dibangun. Pengujian ini dilakukan untuk mengetahui apakah aplikasi penjualan BBM berbasis online telah sesuai dengan kebutuhan pengguna, baik dari aspek kemudahan penggunaan, kesesuaian fitur, kejelasan informasi, maupun manfaat aplikasi dalam mendukung pekerjaan operasional [31], [32], [34].

2.3.1 Database System Development Life Cycle (DBSDLC)

Database System Development Life Cycle (DBSDLC) merupakan metode pengembangan basis data yang dilakukan secara bertahap. Tahapan DBSDLC mencakup perencanaan basis data, definisi sistem, pengumpulan dan analisis kebutuhan, perancangan basis data, pemilihan DBMS, implementasi, serta pengujian [1], [2].

Pada penelitian ini, DBSDLC digunakan untuk merancang basis data terintegrasi pada PT. Indra Perkasa SPBU. Perancangan dilakukan melalui tahapan konseptual, logikal, dan fisik agar struktur basis data yang dihasilkan sesuai dengan kebutuhan aplikasi penjualan BBM berbasis online.

2.3.2 Metode Prototyping

Metode *prototyping* merupakan metode pengembangan sistem yang dilakukan dengan membuat rancangan awal aplikasi sebelum sistem akhir dikembangkan. Rancangan awal tersebut digunakan untuk memberikan

gambaran kepada pengguna mengenai tampilan, alur, dan fungsi aplikasi. Melalui proses evaluasi, pengguna dapat memberikan masukan terhadap *prototype* yang telah dibuat sebelum aplikasi diimplementasikan secara penuh [31], [33].

Pada penelitian ini, metode *prototyping* digunakan untuk membangun aplikasi penjualan BBM berbasis online. Tahapan yang dilakukan meliputi pengumpulan kebutuhan, pembuatan rancangan awal, evaluasi *prototype*, perbaikan *prototype*, pengkodean sistem, pengujian, dan implementasi aplikasi. Metode ini digunakan agar aplikasi yang dibangun dapat disesuaikan dengan kebutuhan pengguna sistem.

2.3.3 User Acceptance Test (UAT)

User Acceptance Test (UAT) merupakan metode pengujian yang digunakan untuk mengetahui tingkat penerimaan dan kepuasan pengguna terhadap sistem yang telah dikembangkan. UAT dilakukan dengan melibatkan pengguna akhir untuk menilai apakah sistem telah sesuai dengan kebutuhan dan dapat digunakan dalam mendukung pekerjaan pengguna [31], [32], [34].

Pada penelitian ini, UAT digunakan untuk mengukur tingkat kepuasan pengguna terhadap aplikasi penjualan BBM berbasis online pada PT. Indra Perkasa SPBU. Pengukuran dilakukan menggunakan kuesioner dengan skala penilaian tertentu. Hasil UAT digunakan untuk mengetahui apakah aplikasi yang dibangun telah sesuai dengan kebutuhan pengguna, terutama dalam pengelolaan data operasional, *monitoring*, dan pelaporan.

2.4 Tools/software yang digunakan

Tools dan *software* digunakan untuk mendukung proses perancangan basis data, implementasi basis data, pengembangan aplikasi penjualan BBM berbasis online, serta penyajian informasi operasional. *Tools* dan *software* yang digunakan dalam penelitian ini meliputi MySQL, phpMyAdmin, Prisma ORM, Next.js, Chart.js, dan Visual Studio Code.

2.4.1 MySQL

MySQL merupakan *Database Management System* (DBMS) yang digunakan untuk menyimpan dan mengelola data dalam bentuk basis data relasional. MySQL mendukung penyimpanan data dalam bentuk tabel, serta memungkinkan hubungan antar data disusun melalui *primary key* dan *foreign key*.

Pada penelitian ini, MySQL digunakan sebagai media penyimpanan basis data untuk aplikasi penjualan BBM berbasis online. Penggunaan MySQL mendukung proses penyimpanan, pembaruan, penghapusan, dan pengambilan data yang dibutuhkan oleh aplikasi.

2.4.2 phpMyAdmin

phpMyAdmin merupakan *tools* berbasis *web* yang digunakan untuk membantu pengelolaan basis data MySQL. *Tools* ini dapat digunakan untuk membuat basis data, melihat struktur tabel, memeriksa data, serta mengelola relasi antar tabel melalui tampilan yang lebih mudah digunakan.

Pada penelitian ini, phpMyAdmin digunakan untuk membantu proses implementasi dan pemeriksaan basis data yang telah dirancang. Penggunaan phpMyAdmin memudahkan proses pengecekan struktur basis data sebelum digunakan pada aplikasi.

2.4.3 Prisma ORM

Prisma ORM merupakan *Object Relational Mapping* yang digunakan sebagai penghubung antara aplikasi dan basis data. Prisma membantu proses akses data melalui model yang disusun sesuai dengan struktur tabel pada basis data.

Pada penelitian ini, Prisma ORM digunakan untuk menghubungkan aplikasi penjualan BBM berbasis online dengan basis data MySQL. Melalui Prisma, proses pengambilan, penambahan, pembaruan, dan penghapusan data dapat dilakukan secara lebih terarah dan konsisten.

2.4.4 Next.js

Next.js merupakan *framework* yang digunakan dalam pengembangan aplikasi berbasis *web*. *Framework* ini mendukung pembuatan halaman aplikasi, pengelolaan tampilan, serta pengembangan fitur yang terhubung dengan data dari basis data.

Pada penelitian ini, Next.js digunakan untuk membangun aplikasi penjualan BBM berbasis online. Aplikasi yang dikembangkan mencakup halaman login, pengelolaan data, laporan, *dashboard*, grafik, dan penyajian informasi operasional lainnya.

2.4.5 Chart.js

Chart.js merupakan *library* yang digunakan untuk menampilkan data dalam bentuk visual, seperti grafik garis, grafik batang, dan diagram lainnya. Penggunaan visualisasi data dapat membantu informasi disajikan secara lebih ringkas dan mudah dipahami.

Pada penelitian ini, Chart.js digunakan untuk mendukung penyajian informasi operasional dalam bentuk *dashboard*. Data yang tersimpan dalam basis data dapat ditampilkan dalam bentuk grafik sehingga pengguna dapat melihat informasi penjualan, stok, pendapatan, biaya, penyusutan BBM, dan laba secara lebih jelas.

2.4.6 Visual Studio Code

Visual Studio Code merupakan *code editor* yang digunakan dalam proses pengembangan aplikasi. *Software* ini mendukung penulisan kode program, pengelolaan *file* proyek, serta penggunaan ekstensi yang dibutuhkan dalam pengembangan aplikasi berbasis *web*.

Pada penelitian ini, Visual Studio Code digunakan untuk mengembangkan aplikasi penjualan BBM berbasis online. Penggunaan *software* ini membantu proses penulisan kode, pengelolaan struktur proyek, serta pengembangan fitur aplikasi.